AF 2123

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re patent application of
Robert J. Munger et al.
Serial No.: 09/468,617
Filed: December 21, 1999
For: METHOD FOR PROGRAMMING OPERATOR SYSTEM INTERFACE WITH A SIMULATOR

Appeal No.:
Group Art Unit: 2123
Examiner: D. Craig

Commissioner for Patents
United States Patent and Trademark Office
P. O. Box 1450
Alexandria, Virginia 22313-1450

BRIEF OF APPELLANTS UNDER 37 C.F.R. 1.192(c)

Sir:

Applicants have, on June 17, 2003, filed a timely Notice of Appeal from the action of the Primary Examiner in finally rejecting claims 1 - 10 in this application. Attached is a check in the amount of $320.00 (37 C.F.R. 1.17(f)) to cover the fee for filing this appeal brief.

REAL PARTY IN INTEREST

The real party in interest in this appeal is Lockheed Martin Corporation of Bethesda, Maryland, assignee of the entire interest of the above-identified application.

RELATED APPEALS AND INTERFERENCES

The Appellants, their legal representative and the assignee are presently unaware of any appeal or interference which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

STATUS OF THE CLAIMS

Claims 1 - 10 stand finally rejected under 35 U.S.C. §103 as being unpatentable over Isreal et al. in view of Klein et al.

STATUS OF AMENDMENTS

All amendments presented during the prosecution of this application have been entered. No amendments were presented subsequent to the final rejection.

SUMMARY OF THE INVENTION

The invention, as claimed, is a method for programming an operator system interface using a simulator rather than the operator interface system itself which may be highly complex and part of a larger system that, in turn, may be incorporated in a complex machine which is expensive and expensive to operate. Further, alteration of control functions of an interface while operating such a machine may present dangers such as operator uncertainty concerning the location of controls, key locations or information or access to it and conversely, the operation of a machine or even the computer system may cause a severe complication in the modification of the interface. Therefore, it invention provides convenience, economy and safety during the development and programming of a complex interface separately from the computer system in which is will principally be used. In other words, the invention provides for the customization of an operator system interface in a manner which is completely decoupled from the system it is used to operate and where it may be

highly undesirable for a facility for interface modification to be provided (e.g. where modification of the locations of controls may cause confusion). By the same token, performing interface modifications in the context of a simulator allows the modifications to be tried and evaluated as they are being made in the simulated interface and for the modified interface to be used for training or demonstration (e.g. as the machine or systems controlled using the interface is modified. The use of the definitional tables of the operator interface system avoids the need for any direct writing or modification of code or separate running of the interface after design or modification. Operation of the invention including an exemplary interface modification operation is detailed at pages 18 - 24 of the specification.

These meritorious functions, summarized on pages 2 - 5 of the specification, are accomplished, as claimed, by using definitional tables from or as would be used in an operational operator system interface, developing a simulator program (e.g. by translation, if needed, from the definitional tables of the operator interface system to the presumably much simpler program of the simulator) which will simulate the operation of the operator system interface such that there is a one-to-one mapping of the operator interface system to the simulator. This correspondence is illustrated in Figure 1 and described on page 6 (e.g. different displays, keys, etc. of the interface are displayed on a desktop computer monitor, an example of which is illustrated in Figure 2 including a multi-function display (MFD) 201, two sets of variable function keys 205a and 205b with variable legend displays and an array of fixed keys 206), modifying the operator system interface as desired (step 304 of Figure 3) and

altering and outputting the definitional tables (step 304 of Figure 3) to correspond to the interface modifications made. These tables are translated, if needed, fed back to the simulator to modify the simulation (as depicted by the arrow from step 305 to step 302 of Figure 3) and/or output for updating of the operational operator system interface (306 of Figure 3) and/or developing system requirements specifications (SRS) tables (since system requirements may not be known or accurately describable during interface development or modification). Thus, in addition to being able to immediately experience the modification through the simulator as the modification is made, when the definitional tables are re-installed in the operator system interface, the operator system interface will be directly reprogrammed to include the changes made on the simulator and documentation developed (e.g. SRS tables) for the modified interface system.

ISSUES

The sole issue in the application is whether or not the claimed subject matter is obvious over Isreal et al. in view of Klein et al. Included sub-issues are whether or not according the claims their broadest reasonable interpretation extends to the effective ignoring of explicit claim recitations and whether or not a *prima facie* demonstration of obviousness has been made.

GROUPING OF CLAIMS

The rejected claims do not stand or fall together. The reasons why appellants consider the rejected claims to be separately patentable are set out in the following section, entitled "ARGUMENT".

ARGUMENT

The Prior Art

Isreal et al. - U. S. 6,330,007

Isreal et al is directed to a prototyping and specification tool for designing a graphical user interface having dynamic keys and dialog boxes using a database in an interactive manner and without requiring the designer to write code by allowing the interface designer to enter design specifications into forms which are stored in a relational database. These specifications cover the visual style of the interface being designed, the specific design of each screen and the navigation among screens (column 2, lines 21 - 23). Once the specifications or a portion thereof have been entered into such forms, the specifications, to the extent they have been entered, can immediately be run as an interactive prototype on the same computer system.

The database structure of Isreal et al. is depicted in Figure 3 and described at column 8, lines 45 - 61, exemplary screens that may be generated are depicted in Figures 8 - 10 and are accessed through the screen depicted in Figure 7 for prototyping. The screens used in the prototyping are depicted in, for example, Figures 11 - 19. Column 7, lines 34 - 40 indicates that the screen of Figure 7 is used *for access to a current screen in the application being prototyped or specified*, refers to it as the "target application" and displays the fixed keys *of the target application*. Thus, in essence, the arrangement of Isreal et al. allows prototyping of an interface for an application which is running on the same machine and allows access to the screens which may be prototyped either through the prototyping program or the running application itself.

Klein et al. - U. S. 5,768,567

Klein et al. is directed to an optimizing hardware/software co-simulator for co-simulation of both hardware and software while minimizing, to the extent possible, the discrepancy in execution time between instruction set simulators and hardware modelers by, for example performing unoptimized memory accesses through hardware simulation while optimized memory accesses are performed by the co-simulation optimization manager. Klein et al. is also directed to validate and discern the operability of the combination of hardware and software components of a system.

The Claimed Invention

The invention as defined by the sole independent claim 1 includes steps of 1.) providing definitional tables which define governing attributes for an operator interface system, 2.) generating a simulator program which simulates the operator interface simulator system; the simulator program, when run on a computing device other than that providing the operator system interface, displays a representation of the operator interface system defined by the definitional tables and allows selection of components of the operator system interface using a pointing device to display information about the selected component and thus to modify the representation of the operator system interface within the simulator, and 3.) modifying the definitional tables to correspond to the modification of the representation of the operator interface system within the simulator to reprogram the operator interface system. Thus, modification of the interface is performed within a simulation of the

interface in a manner totally decoupled from the actual operator system interface, the system and application providing the interface and any apparatus controlled by the application. Moreover, the modifications are performed within the environment of the simulator or simulation itself, as supported by the claimed use of a pointing device for selection and the display of information related to the selection so made and thus do not require a distinct procedural access through a more traditional type of prototyping program to effect a modification.

Dependent claims include the additional functions of generating tables to be used in a software requirements specification (claim 2), generating the operator system interface definitional tables from the simulator (claim 3), extracting definitional tables from the operator system interface (claim 4), updating the definitional tables (claims 5 and 7), repeating steps of claim 1 to update the definitional tables (e.g. to collect proposed interface modifications from a plurality of operators, developing additional modifications and the like - claim 6), running the simulator on a personal computer (claim 8), training operators using the simulator (claim 9) and demonstrating the operator system interface (claim 10).

The Examiner's Application of the Prior Art

In the final rejection of claims 1 - 10, the Examiner first discusses the use of hot keys to switch between programs or macros; referring to Holtz et al. (which is not applied against the claims in the rejection, as stated). The Examiner then admits that Isreal et al. does not teach or suggest use of a simulator program distinct from the operator interface

system but maintains the assertion that all other recitations of the claims are answered by Isreal et al., evidently including the Examiner's position that Isreal et al uses a simulator. The Examiner then applies the Klein et al. reference in combination with Isreal et al. asserting that Klein et al. teaches the simulation of an operator interface program distinct from the operator system interface and that the modification of Isreal et al. would be obvious due to the desirability of simulating hardware in software. Dependent claims are treated merely by citations to passages in Isreal et al. in the initial Office action and not further substantive comments are provided in the final rejection of these claims.

The Differences Between the Prior Art and the Claimed Invention

Initially, the Examiner's assertion that Isreal et al. provides any type of simulation is respectfully but strongly traversed. As pointed out above, Isreal et al. teaches the prototyping of an interface as an "add-on" for a target application which is currently running. Isreal is explicitly clear and distinguishes the actions to be taken in regard to a screen based upon whether or not the screen is accessed through the screen of Figure 7. Column 10, lines 12 - 19, clearly indicate that the screen of figure 7 is used for access to and interaction with a screen of the application through the prototyping tool. otherwise, access to any given screen is through the running application, itself, and no simulation whatsoever, is taught or suggested in Isreal et al. The passage of Isreal et al. at column 2, lines 12 - 26 cited by the Examiner as teaching a simulator is, in fact,

devoid of any such reference but, rather, indicates that a part of the interface of the target application can be used immediately upon the specifications being entered from the forms in the database. Thus the "forms" of Isreal et al. are seen to be much different from the definitional tables as claimed which define the "specific governing attributes of the interface" rather than matters of "style", "design" and "navigation" as noted in the same passage relied on by the Examiner.

Further, by focusing on "hot keys" the Examiner glosses many of the explicit recitations of the "generating" step of claim 1. The simulator of Isreal et al. wherever it is run, does not perform display of a representation of the operator system interface and particularly not in accordance with the definitional tables or allow the user to directly select a component of the interface, much less with a pointing device, to allow viewing information about the selected component or to effect a change in keysets or menus or modify the representation of the interface with the environment of the simulator, itself.

Moreover, in regard to Klein et al., the Examiner merely refers to Figures 2 and 3 a passage from column 1, line 64 to column 2, line 16, which do not contain any reference to an *interface* simulator but only to the considerations and difficulties involved in validating operability of embedded systems through simulation since embedded systems are neither hardware nor software dominant. Therefore, Klein et al. does not appear to teach or suggest the subject matter admitted to be missing from Isreal et al.

In regard to the dependent claims, the final rejection of claims 1 - 10 does not supplement the Examiner's comments from the initial Office action in

which the dependent claims were each treated with a single citation to Isreal et al. In fact, the citations are merely repeated, verbatim, in the final rejection. A comparison of the subject matter of the dependent claims and the cited passages reveals little if any relevance to the claimed subject matter in any of the cited passages. Moreover, since all of the dependent claims are directed to supplementary functions derived through or supportive of the claimed environment of an interface simulator distinct from the operator system interface now admitted by the Examiner to be absent from the teachings of Isreal et al., it is respectfully submitted that each of the dependent claims represents subject matter distinct from and neither taught nor suggested by Isreal et al. and/or Klein et al.

Therefore, in summary, even the combination of Isreal et al. and Klein et al. does not appear to answer *any* of the recitations of the claims and claim 1, in particular. Clearly, the prior art relied upon does not teach or suggest the explicit recitations of the generating or modifying steps of claim 1 which support the meritorious effects of the invention and, while admitting that Isreal et al. does not teach or suggest simulation of an interface on a computing device other that the computing device providing the interface, does not show how Klein et al. teaches any kind of simulation remotely similar to the invention *or* applicable to Isreal et al. or how modification or Isreal et al. in accordance with Klein et al would (or could) be considered to answer the claim recitations. The Examiner's statement of motivation for a modification of Isreal et al. bears no discernible relationship to the actual teaching of Klein et al. or Isreal et al. much less to the invention and the meritorious function produced thereby (e.g. to

decouple interface modification from the operative system to reduce cost and facilitate modification while also incidentally providing enhanced safety, training and interface demonstration). Therefore, the prior art does not lead to an expectation of success in obtaining the meritorious effects of the invention by the method claimed while, to the contrary, since Isreal et al. is directed to prototyping an interface in a manner allowing immediate execution through the target application, any modification to answer the recitations of the claims (or in accordance with the teachings of Klein et al.) would preclude the intended function of Isreal et al. and is thus improper. See In re Gordon et al. 221 USPQ 1125 (Fed. Circ., 1984).

Accordingly, it is clear that the Examiner has not made and, indeed, cannot make a *prima facie* demonstration of obviousness of any claim in the application based on Isreal et al. and Klein et al. and, moreover, the Examiner has not even addressed the principal function of the invention in any way, much less the claimed subject matter which supports that function. Rather, it is respectfully submitted that the3 Examiner has erred in regard to the actual nature of Isreal et al. in asserting that it teaches simulation and filled in the consequent logical gaps (to the small and insufficient extent to which the Examiner has done so toward addressing some few claim recitations) through hindsight or mere terminology out of context (e.g. the mere reference to simulation of a different machine still in the process of design in Klein et al. while the invention is directed to an "operational" interface - see for example, claim 5 - or where the governing attributes of the interface system are defined by definitional tables as recited in claim 1).

CONCLUSION

For the foregoing reasons, it is respectfully submitted that the rejection of claims 1 - 10 under 35 U.S.C. §103 based on Isreal et al and Klein et al. is clearly in error. The Examiner has not made a *prima facie* demonstration of how the teachings, suggestions or evidence of the level of ordinary skill in the art would or could answer the specific recitations of the claims or support a conclusion of obviousness in regard to the subject matter of the claims or, for that matter, even the concept of the meritorious function of the invention. The Examiner has not properly analyzed the prior art under Graham v. John Deere, 148 USPQ 459 (1966) to determine the scope and content of the prior art or the differences between the prior art and the subject matter of the claims or the level of ordinary skill in the art. but rather, has erred through hindsight and compounded that error through further hindsight and a lack of understanding of either the claimed subject matter or the prior art relied upon resulting in an improper combinations of teachings which, in any case, fail to answer the explicit recitations of the claims.

Accordingly, it is respectfully submitted that the rejection of claims 1 - 10 is in clearly in error and unsupported by the prior art relied upon by the Examiner while the Examiner has failed to make a *prima facie* demonstration of obviousness of any claim in the application. Therefore, reversal of the Examiner's final rejection of claims 1 - 10 is respectfully requested.

If an extension of time is required for this response to be considered as being timely filed, a conditional petition is hereby made for such extension of time. Please charge any deficiencies in fees and credit

any overpayment of fees to Attorney's Deposit Account No. 50-2041.

Respectfully submitted,

Marshall M. Curtis
Registration No. 33,138

Whitham, Curtis, & Christofferson
11491 Sunset Hills Rd., Suite 340
Reston, Virginia 20190
(703) 787-9400

Customer Number: **30743**

APPENDIX

THE CLAIMS ON APPEAL

The claims on appeal are as follows:

1. (Amended) A method for programming an operator system interface with a simulator, said method comprising the steps of:

providing definitional tables for an operator system interface, wherein said tables define specific governing attributes of said operator system interface;

generating an operator system interface simulator program distinct from said operator system interface, wherein, when the simulator program is run on a computing device other than a computing device providing said operator system interface, said simulator program performs display of a representation of the operator system interface defined by the definitional tables input in the providing step and allows a user to select components of the operator system interface, using a pointing device, in order to view information about the selected component on a display device or to effect a change in keysets or menus, thereby modifying said representation of the operator system interface within said simulator program, and

modifying said definitional tables to correspond to said modifying of said representation to reprogram said operator system interface.

2. (Original) A method as recited in claim 1, further comprising the step of generating tables to be used in a software requirements specification.

3. (Original) a method as recited in claim 1, further comprising the steps of:

generating operational operator system interface definitional tables using the simulated operator system interface definitional tables; and

developing an operational operator system interface from th generated operational operator interface definitional tables.

4. (Original) A method as recited in claim 1, wherein the providing step further comprises the step of extracting the definitional tables from an existing operator system interface.

5. (Amended) A method as recited in claim 4, further comprising the step of:

generating updated operational operator system interface definitional tables.

6. (Original) A method as recited in claim 4, further comprising the steps of:

modifying the simulated operational operator system interface; and

generating updated operational operator system interface definitional tables.

7. (Original) A method as recited in claim 5, wherein the steps of generating a simulated operator system interface simulator program, modifying the simulated operator system interface program and generating updated operational operator system interface definitional tables are repeated a desired number of times.

8. (Original) A method as recited in claim 1, further comprising the step of running the simulator program on a personal computer.

9. (Original) A method as recited in claim 8, wherein the simulator program is used to train operators in a control and display system defined by the operator system interface.

10. (Original) A method as recited in claim 8, wherein the simulator program is used to demonstrate functionality of a control and display system defined by the operator system interface.